

1 CLAIMS:

2 What is claimed is:

3 1. A method, comprising:

4 (a) maintaining a device identifier and a private key
5 in a programmable logic device, the device identifier and
6 the private key being non-volatile such that if power to
7 the programmable logic device is lost the device
8 identifier and private key remain stored in the
9 programmable logic device;10 (b) receiving a first encrypted key onto the
11 programmable logic device, and using the device
12 identifier and the private key to decrypt the first
13 encrypted key thereby generating a first key;14 (c) receiving onto the programmable logic device a
15 bitstream comprising first encrypted configuration data
16 encrypted with the first key;17 (d) using the first key to decrypt the first encrypted
18 configuration data on the programmable logic device
19 thereby generating first configuration data; and20 (e) configuring a first portion of the programmable
21 logic device using the first configuration data.22 2. The method of Claim 1, wherein neither the device
23 identifier nor the private key are rewritable.24 3. The method of Claim 1, wherein the bitstream further
25 comprises a first key number associated with the first
26 encrypted configuration data, the first key being stored
27 on the programmable logic device in association with the
28 first key number, the programmable logic device in step
29 (d) using the first key number in the bitstream to

1 identify the first key as the key that will be used in
2 step (d) to decrypt the first encrypted configuration
3 data.

4

5 4. The method of Claim 1, wherein the device identifier
6 and the private key are stored on the programmable logic
7 device in one of the group consisting of: an antifuse-
8 based storage element, a fuse-based storage element, a
9 laser-programmed storage element, an EPROM storage
10 element, and a flash-based storage element.

11

12 5. The method of Claim 1, further comprising:
13 after the first key is generated in step (b), storing
14 the first key in non-volatile memory on the programmable
15 logic device.

16

17 6. The method of Claim 1, wherein the first encrypted
18 configuration data is decrypted in step (d) on the
19 programmable logic device by a hardware decryptor.

20

21 7. The method of Claim 1 wherein
22 step (b) further comprises receiving a second encrypted
23 key onto the programmable logic device and using the
24 device identifier and the private key to decrypt the
25 second encrypted key, thereby generating a second key;
26 step (c) further comprises receiving onto the
27 programmable logic device a bitstream comprising second
28 encrypted configuration data encrypted with the second
29 key;
30 step (d) further comprises using the second key to
31 decrypt the second encrypted configuration data on the

1 programmable logic device, thereby generating second
2 configuration data; and
3 step (e) further comprises configuring a second
4 portion of the programmable logic device using the second
5 configuration data.

6

7 8. The method of Claim 7, wherein the bitstream further
8 comprises a first key number associated with the first
9 encrypted configuration data, and wherein the bitstream
10 further comprises a second key number associated with the
11 second encrypted configuration data, the first key being
12 stored on the programmable logic device in association
13 with the first key number, the second key being stored on
14 the programmable logic device in association with the
15 second key number, the programmable logic device in (d)
16 using the first key number in the bitstream to identify
17 the first key as the key that will be used in (d) to
18 decrypt the first encrypted configuration data, the
19 programmable logic device in (d) using the second key
20 number in the bitstream to identify the second key as the
21 key that will be used in (d) to decrypt the second
22 configuration data.

23

24 9. The method of Claim 7 further comprising:
25 after the first key and the second key are generated in
26 step (b), storing the first key and the second key in
27 non-volatile memory on the programmable logic device.

28

29 10. The method of Claim 7, wherein the first portion of
30 the programmable logic device is configured in (e) to
31 realize a first IP module, and wherein the second portion

1 of the programmable logic device is configured in (e) to
2 realize a second IP module.

3

4 11. The method of Claim 1, wherein the programmable logic
5 device is an SRAM-based PLD.

6

7 12. The method of Claim 10, wherein the non-volatile
8 memory in the programmable logic device is flash-based.

9

10 13. The method of Claim 10, wherein the non-volatile
11 memory in the programmable logic device is one-time
12 programmable.

13

14 14. The method of Claim 10, wherein the non-volatile
15 memory in the programmable logic device is antifuse-
16 based.

17

18 15. The method of Claim 10, wherein the non-volatile
19 memory in the programmable logic device is fuse-based.

20

21 16. The method of Claim 1, wherein the device identifier
22 and the private key are rewritable at one time, but as of
23 the time step (a) occurs are no longer rewritable.

24

25 17. The method of Claim 1, further comprising:
26 receiving on a license manager the device identifier
27 maintained on the programmable logic device;
28 receiving on the license manager a first authorization
29 code; and
30 determining whether the first authorization code has a
31 predetermined relationship with respect to the device
32 identifier, wherein if the first authorization code is

1 determined to have the predetermined relationship then
2 the license manager sends the first encrypted key to the
3 programmable logic device such that it is received in
4 step (b), and wherein if the first authorization code is
5 determined not to have the predetermined relationship
6 then the license manager does not send the first
7 encrypted key to the programmable logic device in step
8 (b)

9

10 18. The method of Claim 17, wherein the first
11 authorization code has the predetermined relationship
12 with respect to the device identifier if the first
13 authorization code contains the device identifier in an
14 encrypted form.

15

16 19. A method comprising:
17 receiving onto a programmable logic device an
18 encrypted first key;
19 on the programmable logic device decrypting the
20 encrypted first key to generate a first key and storing
21 the first key on the programmable logic device;
22 receiving onto the programmable logic device a
23 configuration bitstream having a first portion and a
24 second portion;
25 on the programmable logic device decrypting the first
26 portion of the configuration bitstream using the first
27 key;
28 configuring the programmable logic device with the
29 decrypted first portion of the configuration bitstream
30 thereby realizing a first IP module;.

31

1 20. A programmable logic device that receives an encrypted
2 configuration bitstream, the programmable logic device
3 comprising:

4 non-volatile storage that stores a first key;
5 a decryptor that decrypts a first part of the
6 encrypted configuration bitstream using the first key and
7 thereby generates first configuration data; and
8 first configurable logic elements being configured by
9 the first configuration data.

10

11 21. The programmable logic device of Claim 20, wherein the
12 first part of the encrypted configuration bitstream is
13 identified by a first key number in the bitstream, the
14 first key number identifying the first key in the non-
15 volatile storage.

16

17 22. A method, comprising:
18 receiving on a development system a device identifier
19 from a programmable logic device;
20 receiving on the development system an authorization
code;
22 verifying on the development system that the
23 authorization code and the device identifier have a
24 predetermined relationship, wherein if the authorization
25 code and the device identifier have the predetermined
26 relationship then encrypting a key using the device
27 identifier and sending the encrypted key from the
28 development system to the programmable logic device, but
29 wherein if the authorization code and the device
30 identifier do not have the predetermined relationship
31 then the encrypted key is not sent from the development
32 system to the programmable logic device; and

1 the development system using the key to encrypt a
2 portion of a configuration data bitstream, the
3 development system outputting the configuration data
4 bitstream including the encrypted portion.

5

6 23. The method of Claim 22, wherein the key has a key
7 number, and wherein the development system adds the key
8 number to the configuration data bitstream such that the
9 key number is associated with the encrypted portion of
10 the configuration data bitstream, the configuration data
11 bitstream output from the development system including
12 the encrypted portion and the key number.

13

14 24. The method of Claim 22, wherein the development system
15 comprises a capture/design tool and a license manager,
16 the method further comprising:

17 if the authorization code and the device identifier
18 are verified as having the predetermined relationship
19 then the license manager allows use of IP module design
20 information by the capture/design tool, whereas if the
21 authorization code and the device identifier are not
22 verified as having the predetermined relationship then
23 the license manager does not allow use of the IP module
24 design information by the capture/design tool.

25

26 25. The method of Claim 22, wherein the portion of the
27 configuration data bitstream is configuration data for an
28 IP module, the development system comprising a
29 capture/design tool, the capture/design tool being usable
30 to view a net external to the IP module, the
31 capture/design tool not being usable to view a net
32 internal to the IP module.

1
2 26. A development system, comprising:
3 a capture/design tool; and
4 means for verifying that an authorization code has a
5 predetermined relationship with respect to a device
6 identifier read from a programmable logic device, and if
7 the authorization code is verified then the means also
8 encrypting a key and sending the encrypted key to the
9 programmable logic device, if the authorization code is
10 verified then the means also uses the key to encrypt a
11 portion of a configuration data bitstream output by the
12 capture/design tool, the configuration data bitstream
13 including the encrypted portion being sent to the
14 programmable logic device.

15
16 27. The development system of Claim 26, wherein the
17 encrypted portion of the bitstream contains configuration
18 data for an IP module, the capture/design tool being
19 usable to view a net external to the IP module, the
20 capture/design tool being unusable to view a net internal
21 to the IP module.

22
23 28. The development system of Claim 26, wherein the key
24 has a key number, the means inserting the key number into
25 the configuration data bitstream sent to the programmable
26 logic device, the key number in the configuration data
27 bitstream being associated with the encrypted portion of
28 the configuration data bitstream.

29